ROI Measurement for Test Automation Projects
A Case Study

Table of Contents

1. Introduction
2. Why Should We Measure ROI?
3. How Should We Measure ROI?
4. Steps to Improve ROI
5. Case Study
6. Conclusion
7. Contributions

Abstract

In light of software development's fast-paced nature, test automation is crucial to ensure high-quality software development. Test automation also helps expedite business decision-making and allows resources to focus on areas requiring strategic attention. Test automation can be justified by measuring the Return on Investment (ROI). The paper presents a case study of a project where test automation was implemented to improve the ROI.

Introduction

Test automation is a key feature in ensuring continuous testing along with the development. It should aim at being scalable and reduce maintenance efforts in a world of demanding business needs, evolving requirements, changing technology interfaces and people.

Why Should We Measure ROI?

Measuring anything will help us in identifying the efficiency and effectiveness of any investment. Also, how and by how much we are benefiting is important and measuring ROI helps in identifying it.

How Should We Measure ROI?

Calculating return on investment (ROI) on automation projects is an easy affair if we know what are the efforts that are being put into it. The two inputs required are the total cost and the total time spent on the project.

Steps to Improve ROI

Test automation can be improved by improving automation framework and scripting. Well-structured and reusable keywords make the framework more flexible, easy to maintain, and scalable. Framework should provide a set of reusable keywords that would enable the engineer to consume them without having to come up with completely new scripts.

Conclusion

ROI measurement is a key performance indicator in automated testing. Moving to Test automation ROI, it's a key performance indicator in automated testing. To understand if test automation is justified, test cases should be prioritized by business value. Automated tests should be run in parallel with manual tests to ensure that the business value metrics are being met. It is highly recommended to measure the ROI to reduce the cost and time as far as possible.
We have worked on a project where we started from developing the framework using open source tools by integrating open source external reports into the tool. Over a period of 3 years team has developed 1600+ scripts and are maintaining them. These test scripts are executed for every sprint which is 14 days in duration. And these scripts are also executed for ad hoc release requests.

### Calculating ROI

Since ROI is a unitless number, it really does not matter whether the savings and investment amounts are in dollars or time. For ease in calculation, hours will be used because most of our inputs are in the form of time.

Before we actually start calculating the ROI, we need to consider all the factors which affect the ROI.

#### Case Study

Calculating ROI

Since ROI is a unitless number, it really does not matter whether the savings and investment amounts are in dollars or time. For ease in calculation, hours will be used because most of our inputs are in the form of time.

Before we actually start calculating the ROI, we need to consider all the factors which affect the ROI.

Here we would like to segregate all the factors into 3 different areas mentioned below:

1. **Initial time for development setup**
2. **Actual Development time**
3. **Monitoring and Maintenance time**

### Initial time for development setup

The pre-phase of the development activity involves parameters like setting up the automation tool, setting up different environments, exploring the applications, time spent for initial data setup etc.

### Actual Development time

In this phase, all the development activities will take place. Parameters like the average number of scripts developed per day and average time to develop each script, code push and sanity check, code review, etc., can be considered to calculate the actual development time. For a manual testing one can consider average scenarios authored per day, average time to author each scenario, total time to review the test case document etc. can be considered to calculate the actual development time.

### Monitoring and Maintenance time

This is the phase where automation saves a lot of time compared to manual testing. All maintenance-related activities can be considered as parameters to calculate the actual number of monitoring and maintenance hours.

### Cost savings

Savings is the difference between the cost of running a set of tests manually versus running the same tests automatically several times over some period.

Over 3 years, 72 regression run cycles we gain around 2000+ hours when compared to manual testing.

Let us consider another scenario, where after one year, the project ended or the test automation framework is abandoned; in either case, it would have been more cost-effective to go with manual testing.

### Conclusion

ROI metrics is important to gain visibility on the success of test automation even before we start to resize this will help in making strategic decisions.

Get in touch with us for feasibility studies, joint workshops, pilots & ideation sessions. Let's connect